Vlasov Simulation Methods for plasmas and beams II

Eric Sonnendrücker
IRMA, Université Louis Pasteur
Strasbourg, France
Co-workers: F. Filbet, M. Gutnic, I. Paun,
N. Besse, J.-L. Lemaire (CEA)

Berkeley, July 31, 2002

Outline

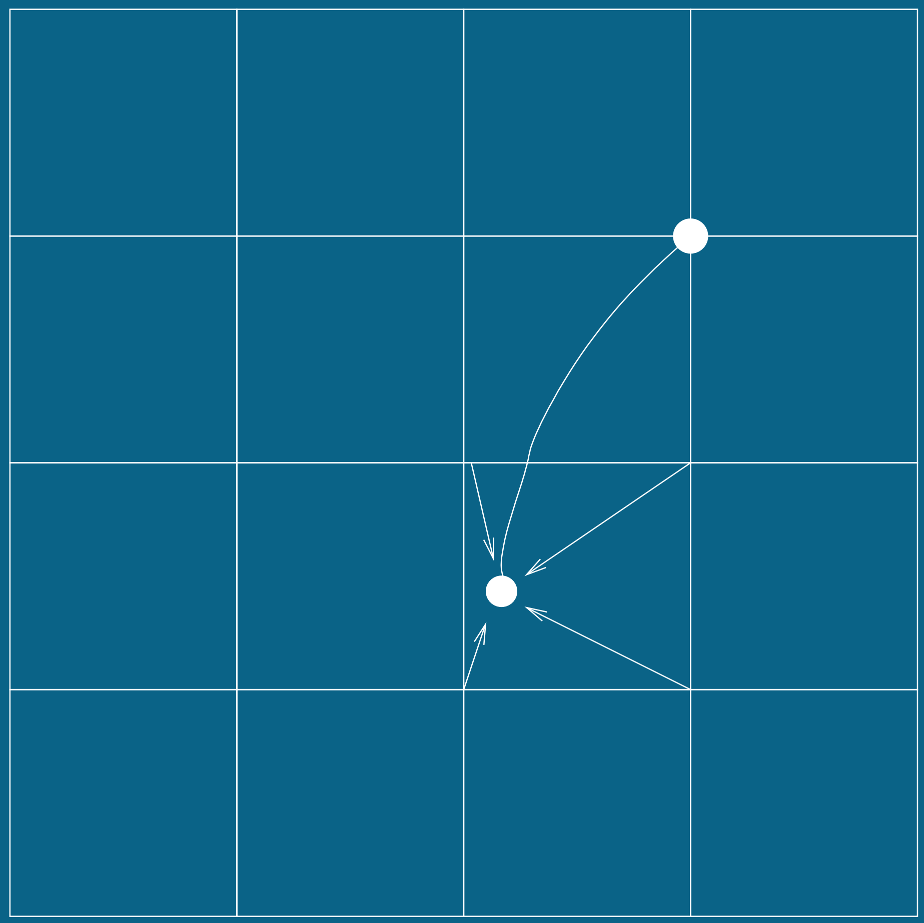
- The axisymmetric Vlasov equation.
- Semi-Lagrangian methods on unstructured meshes.
- An adaptive method based on a wavelet decomposition.

Last week

- Overview of Vlasov method.
- In particular: The semi-Lagrangian method (forward and backward).
- Applications for beam propagation in uniform and periodic focusing channels.

The backward semi-Lagrangian Method

- f conserved along characteristics
- Find the origin of the characteristics ending at the grid points
- Interpolate old value at origin of characteristics from known grid values $\longrightarrow$ High order interpolation needed



The axisymmetric Vlasov equation

The distribution function $f(r, v_r, v_\theta, t)$ is given by

$$\frac{\partial f}{\partial t} + v_r \frac{\partial f}{\partial r} + v_z \frac{\partial f}{\partial z} + \left(\frac{qE_s r}{m} + \frac{q\, B_z}{m} v_\theta + \frac{v_\theta^2}{r} \right) \frac{\partial f}{\partial v_r}$$
$$- \left(\frac{q\, B_z}{m} v_r + \frac{v_\theta\, v_r}{r} \right) \frac{\partial f}{\partial v_\theta} + \frac{qE_s z}{m} \frac{\partial f}{\partial v_z} = 0,$$

where B_z is external and E_s given by Poisson's equation

$$\nabla \cdot E = \rho(t, r)/\varepsilon_0, \quad \rho(t, r) = q \int_{\mathbb{R}^3} f\, dv.$$

Invariants of the Vlasov equation

In order to reduce the dimension of the problem we use the invariance of the canonical angular momentum

$$P(r, v_\theta) = m r v_\theta + \frac{r^2}{2} q B_z.$$

Denoting by $I = \frac{P}{m}$ and making the change of variable $(r, v_r, v_\theta) \longrightarrow (r, v_r, I)$ with $v_\theta = \frac{I}{r} - \frac{1}{2}\frac{q\,B_z}{m} r$, we get

$$\frac{\partial f}{\partial t} + v_r \frac{\partial f}{\partial r} + \left(\frac{q}{m} E_s(t,r) + \frac{I^2}{r^3} - \frac{1}{4}\left(\frac{q\,B_z}{m}\right)^2 r \right) \frac{\partial f}{\partial v_r} = 0.$$

Discretization of the axisymmetric Vlasov equation

- Invariant I is a parameter but needs careful discretization. Characteric curves of the form

$$\frac{\omega^2}{2} r^2 + v_r^2 + \frac{I^2}{r^2} = const.$$

$\longrightarrow$ necessary to control I/r hence I is discretized according to $I = \omega\, r^2$, in vicinity of axis.

- Difficulty near $r = 0$ because of I^2/R^3 term.

- Time-Splitting scheme:
  - ⋆ Advection in r : $\frac{\partial f^*}{\partial t} + v_r \frac{\partial f^*}{\partial r} = 0$,
  - ⋆ Advection in v_r :

$$\frac{\partial f^{**}}{\partial t} + \left(\frac{q}{m} E_s(t,r) + \frac{I^2}{r^3} - \frac{1}{4} \left(\frac{q\,B_z}{m} \right)^2 r \right) \frac{\partial f^{**}}{\partial v_r} = 0.$$

- Cubic Hermite interpolation with numerical computation of derivatives by a fourth order finite difference scheme

$$\partial_r f_i^n = \frac{1}{12 \Delta r} \left[8 \left[f_{i+1}^n - f_{i-1}^n \right] - \left[f_{i+2}^n - f_{i-2}^n \right] \right].$$

Parallelization

- Straightforward using the Invariant.
- Only communications are for gathering of ρ.

# CPU	2D Cartesian PFC	Axisymmetric
4	178 min	59 min
8	89 min	27 min

Table 1: *Computational time for a 2D$\times$ 2D cartesian and axisymmetric solvers.*

The semi-Lagrangian method on unstructured grids

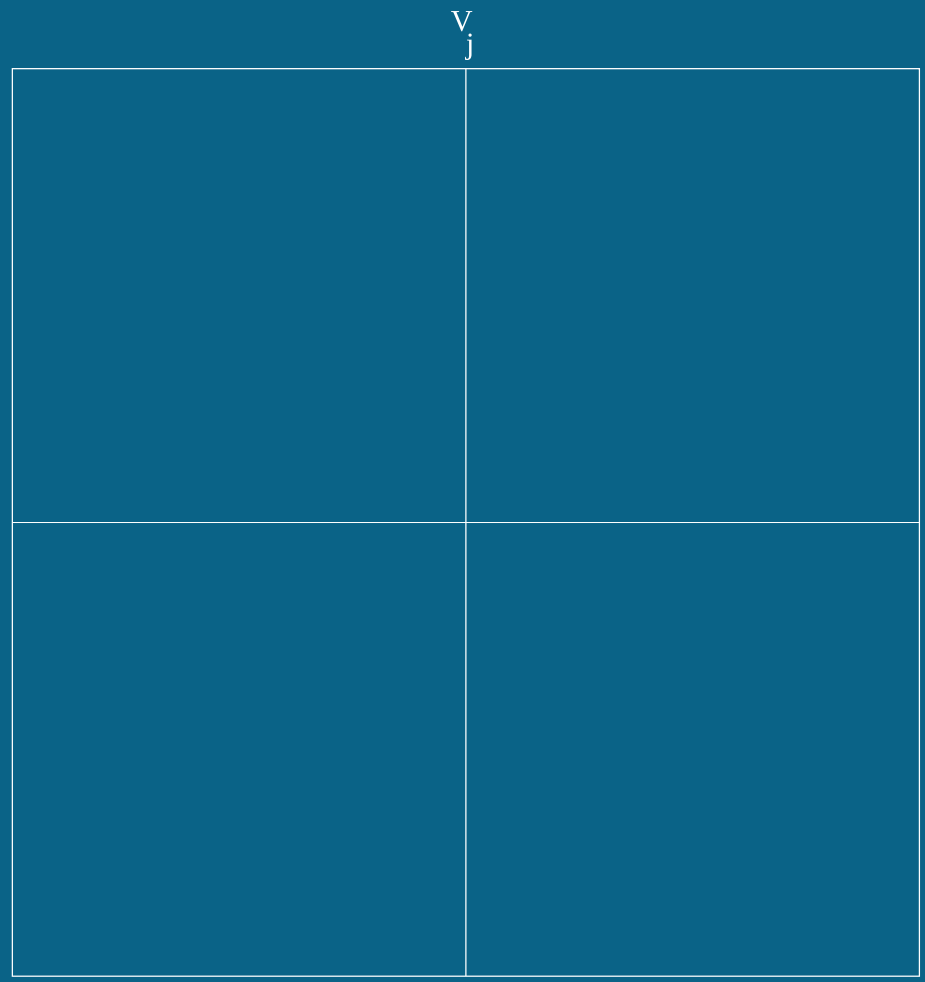
- Give more flexibility to mesh.
- Operator splitting used between x and v advections so that advection is explicit.
- Interpolation in full phase space necessary at each step.
- Interpolation done using Finite Element basis functions $\longrightarrow$ many possibilities.
- Computation of $\rho(x)$ needs interpolations.

Properties of scheme

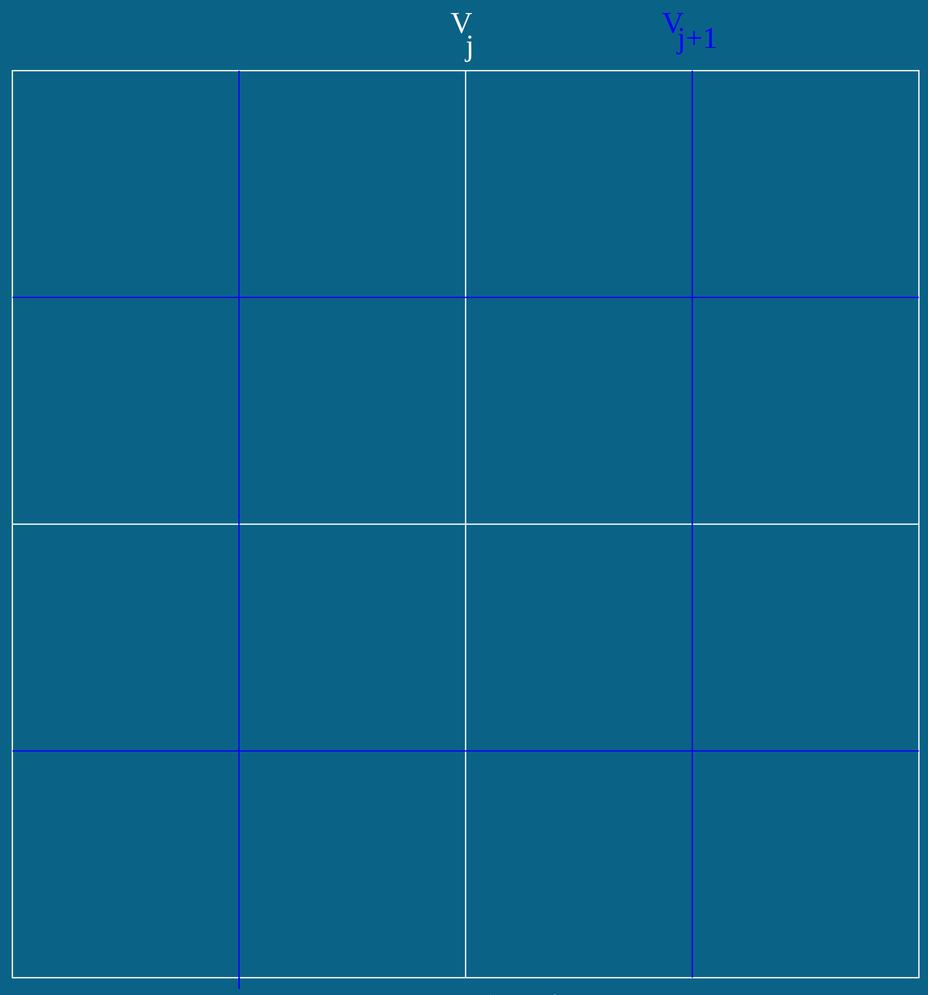
- High order Lagrange seems unstable due to oscillations on edge of elements.
- Hermite type elements with CIP type method seems the best choice in this case.
- Positiveness and conservativeness can be ensured by switching between high order and first order interpolation.

Adaptive semi-Lagrangian method

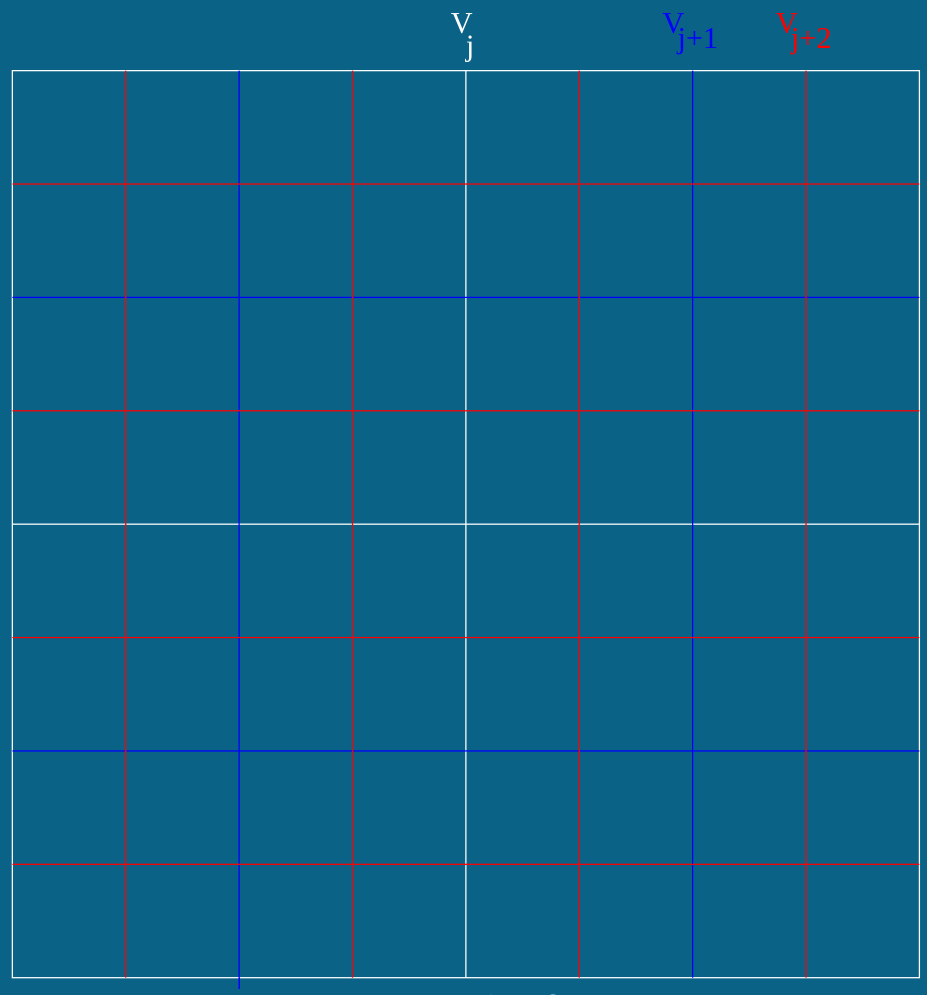
- We want to optimize the number of grid points for a given numerical error.
- Multi-resolution techniques using interpolating wavelets are well suited to determine where refinement is needed.
- Principle of the method
  - ⋆ Use different levels of meshes
  - ⋆ At one given level, decompose gridfunction into gridfunction at coarser level + details.



Grid G_j, grid points $x^j_k = k\,2^j$, level j



Grid G_{j+1}, grid points $x_k^{j+1} = k\,2^{j+1}$, level $j+1$



Grid G_{j+2}, grid points $x_k^{j+2} = k\,2^{j+2}$, level $j+2$

The wavelet decomposition

- Idea: Decompose more precise sample, i.e. values of f at grid points of G_{j+1} (denoted by c_{i+1}) into smaller sample i.e. values of f at grid points of G_j (denoted by c_i) + details (denoted by d_i).
- Details contain difference between exact value and value predicted using interpolation operator.

$$c_{2k}^{j+1} = c_k^j \quad \text{same value at coarse mesh points}$$

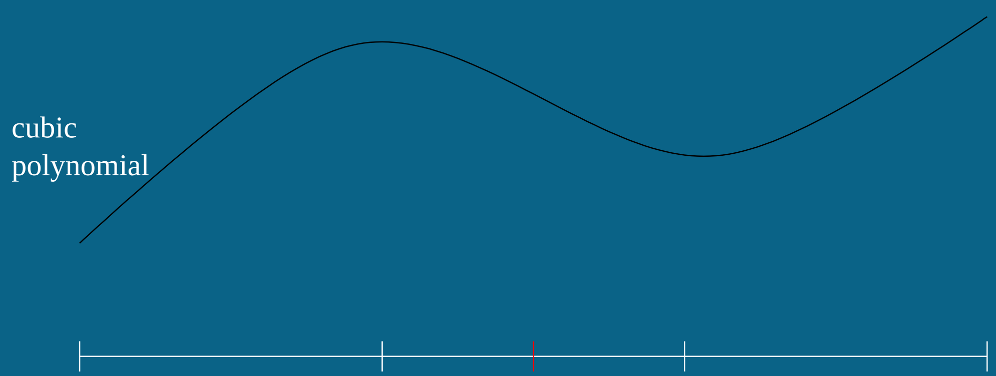
$$d_k^j = c_{2k+1}^{j+1} - P_{2N+1}(x_{2k+1}^{j+1}).$$

Prediction operator

Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.

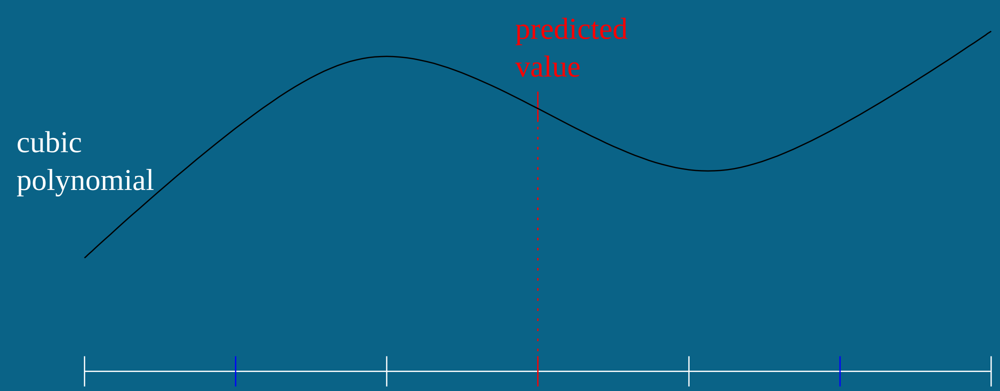
Prediction operator

Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.



Prediction operator

Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.

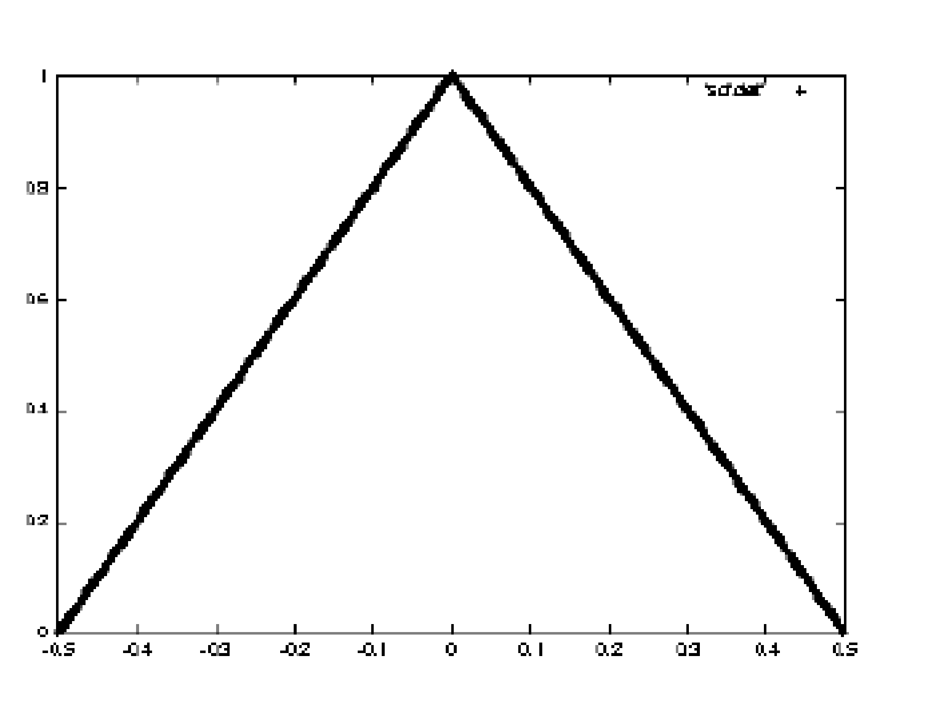


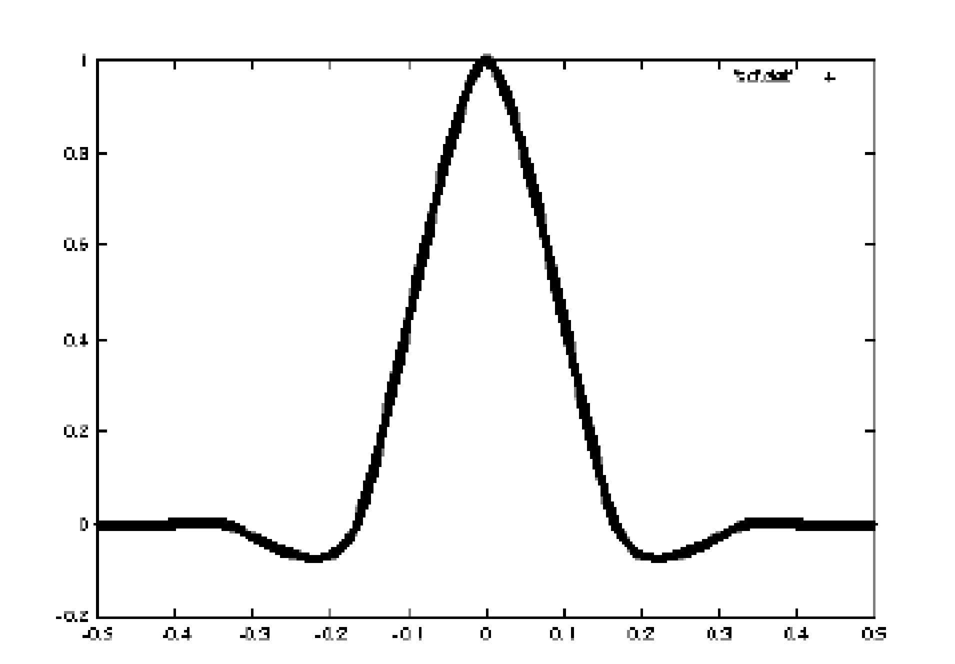
$$d_k^j = c_{2k+1}^{j+1} - P_{2N+1}(x_{2k+1}^{j+1}) \text{ and } c_{2k}^{j+1} = c_k^j$$

Adaptivity and semi-Lagrangian method

- Semi-Lagrangian method based on polynomial interpolation.
- Main idea for adaptivity: details are small where interpolation does a good job.
- In adaptive method, use wavelet decomposition to eliminate grid points corresponding to small details.
- No loss of information due to wavelet decomposition.

Scaling function





Wavelet interpolation

Interpolation formula

$$f^*(x,v) = \sum_{k_1,k_2} \Bigg(c^{j_0}_{k_1,k_2}\, \varphi^{j_0}_{k_1}(x)\, \varphi^{j_0}_{k_2}(v) + \sum_{j_0}^{j_1-1} \Big(d^{row,j}_{k_1,k_2}\, \psi^{j+1}_{k_1}(x)\, \varphi^{j}_{k_2}(v) + d^{col,j}_{k_1,k_2}\, \varphi^{j}_{k_1}(x)\, \psi^{j}_{k_2}(v) + d^{mid,j}_{k_1,k_2}\, \psi^{j+1}_{k_1}(x)\, \psi^{j+1}_{k_2}(v) \Big) \Bigg) \quad (1)$$

The Algorithm for the Vlasov Problem...

- **Initialisation**: decomposition and compression of f_0.
- **Prediction in x** of the grid $\tilde{G}$ (for important details) at the next split time step following the characteristics forward. Retain points at level just finer.
- **Construction of $\hat{G}$**: grid where we have to compute values of f^* in order to compute its wavelet transform.

...The Algorithm for the Vlasov Problem...

- **Advection-interpolation in** x: follow the characteristics backwards in x and interpolate using wavelet decomposition (1): $f^*(x, v) = f^n(x - v\,\Delta t, v)$
- **Wavelet transform of** f^*: compute the c_k and d_k coefficients at the points of $\tilde{G}$.
- **Computation** of electric field from Poisson.
- **Prediction in** v : as for x.

...The Algorithm for the Vlasov Problem

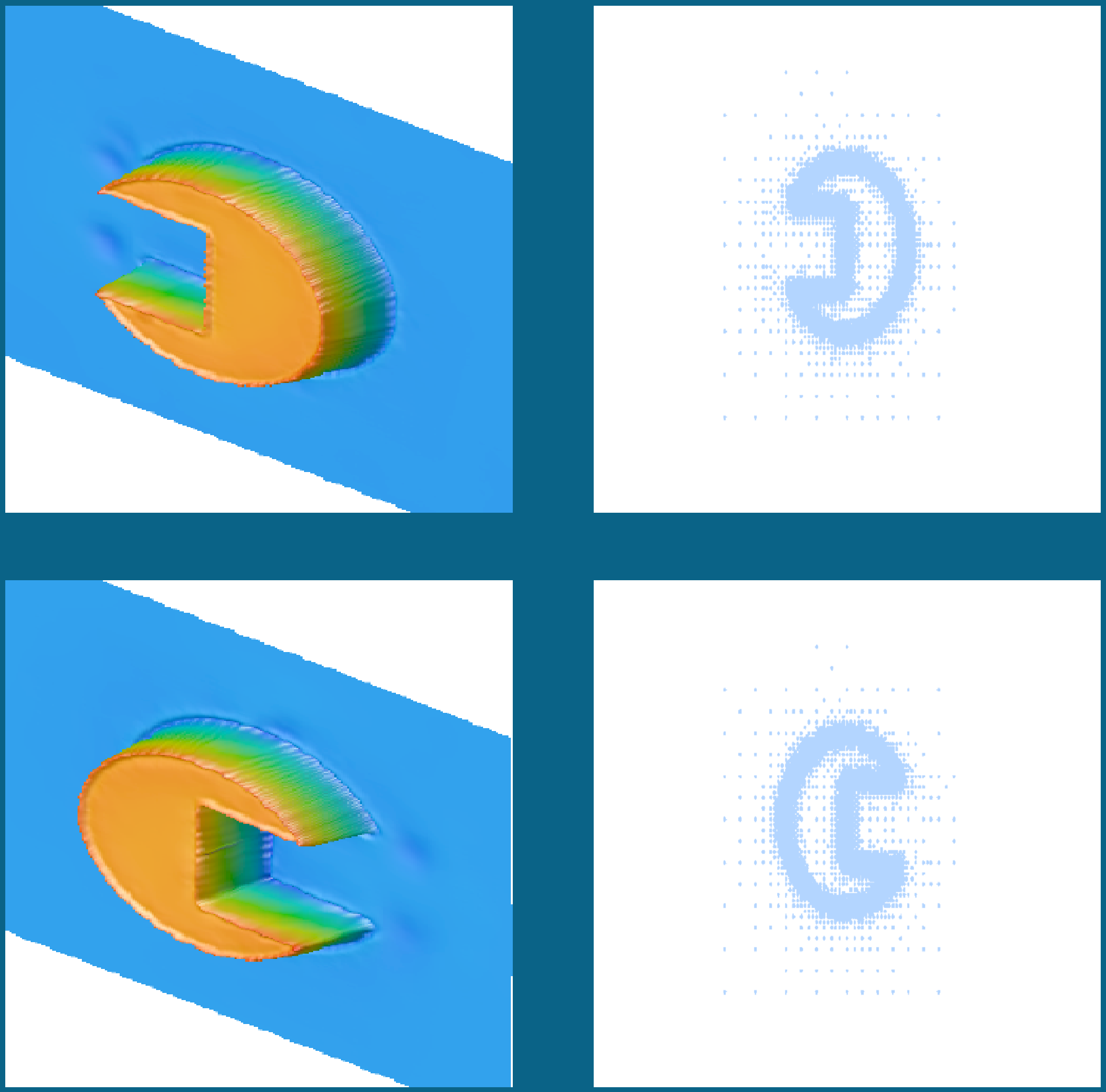
- **Construction of $\hat{G}$**: grid where we have to compute values of f^{n+1} in order to compute its wavelet transform.

- **Advection-interpolation in v**: as for x $f^{n+1}(x,v) = f^*(x, v - E(x)\,\Delta t)$ using wavelet decomposition.

- **Wavelet transform of f^{n+1}**: compute the c_k and d_k coefficients at the points of $\tilde{G}$.

- **Compression of f^{n+1}**.

Numerical results



A few papers and web pages

- Nicolas Besse : Semi-Lagrangian schemes for the Vlasov equation on an unstructured mesh of phase space
  http://www-irma.u-strasbg.fr/irma/publications/2002/02028.shtml
- Francis Filbet, Jean-Louis Lemaire, Eric Sonnendrucker : Direct axisymmetric Vlasov simulations of space charge dominated beams
  http://www-irma.u-strasbg.fr/irma/publications/2002/02009.shtml
- Francis Filbet, Eric Sonnendrucker : Comparison of Eulerian Vlasov Solvers

http://www-irma.u-strasbg.fr/irma/publications/2001/01035.shtml

- Francis Filbet, Eric Sonnendrucker, Pierre Bertrand: Conservative numerical schemes for the Vlasov equation. J. Comp. Phys. Volume 172, Number 1 pp. 166-188 (2001).
- The VADOR code: `http://www-irma.u-strasbg.fr/~filbet`

Conclusions

- Axisymmetric problems can be brought back to 4D phase space or 2D slice which makes them tractable thanks to use of invariance of angular momentum.
- Unstructured meshes feasible but more complicated and fairly slower.
- Adaptive method looks promising in 2D phase space. Implementation more complex due to adaptive mesh structure. Needs to be extended to higher dimensions.